PROJECT ABSTRACT

ESP32-CAM PPE Detection for Construction Sites

YOLOv8-based helmet-compliance monitor: ESP32-CAM nodes stream site frames to a server-side detector with a live dashboard, delivered as a built-to-order student project with firmware, training pipeline, report, PPT and viva kit.

01. Title

ESP32-CAM PPE Detection for Construction Sites

02. Introduction / Background

Personal protective equipment compliance on construction sites is still enforced by supervisors walking the site, a method that cannot watch every corner continuously and leaves no auditable record of violations. Single-stage object detectors of the YOLO family now run fast enough to screen site imagery in near real time, and the ESP32-CAM puts a Wi-Fi camera node in the student budget range. This project combines the two honestly: the ESP32-CAM captures and streams frames only, while a YOLOv8s detector fine-tuned on annotated construction-site imagery runs on the site PC or laptop, flagging workers without helmets with bounding boxes, confidence scores and a live compliance dashboard.

03. Problem Statement

Manual helmet-compliance checks do not scale across large sites, violations go unnoticed, and incident reviews have no evidence trail. The project addresses this by building an automated vision loop: cheap camera nodes cover multiple zones, a trained detector classifies helmet usage per worker, and every violation is logged with timestamp, camera ID, confidence and the evidence frame for review by the site engineer.

04. Objectives

- Build ESP32-CAM firmware that captures site frames and streams them over Wi-Fi from multiple registered nodes.
- Fine-tune YOLOv8s on a Hard Hat Workers-style annotated dataset for helmet / no-helmet / person classes, with a design target of mAP@0.5 around 0.85 measured on the student's own validation split.

- Build a server pipeline (frame ingestion, inference, WebSocket feed) and a dashboard with annotated live feed, compliance statistics, violation log and trend charts.
- Deliver the complete student kit: firmware, training notebooks, dashboard code, deployment guide, report, PPT and viva Q&A.

05. Proposed Methodology

The work proceeds in four stages. (1) Hardware: ESP32-CAM nodes are assembled, flashed with the capture/streaming firmware and positioned to cover the demonstrator site zones. (2) Data: the annotated dataset is analyzed for class balance, split into train/validation/test, and augmented (mosaic, HSV jitter, flips). (3) Training: YOLOv8s pre-trained on COCO is fine-tuned for about 50 epochs; every epoch logs mAP@0.5, precision, recall and loss curves, and the best checkpoint is kept. (4) Integration: the server pulls frames from all nodes, runs inference with non-maximum suppression, overlays labelled boxes, and streams results and violation events to the dashboard.

06. System Architecture / Working

ESP32-CAM nodes capture JPEG frames at a configured interval and serve them over Wi-Fi. The Flask/FastAPI server polls each registered node, batches frames for the YOLOv8s model (640x640 inference), filters detections by a configurable confidence threshold, and draws labelled boxes. Annotated frames and per-camera compliance rates are pushed to the browser over WebSockets; no-helmet detections are written to a SQLite violation log with the evidence frame. A nightly job aggregates detections into daily compliance trends and per-zone summaries. Inference stays on the server because the ESP32-CAM cannot run a YOLO model - the report states this split explicitly.

07. Hardware / Software Requirements

- ESP32-CAM (AI-Thinker) nodes, USB-serial programmer, 5 V supplies
- Site PC/laptop for inference (CPU workable, GPU optional)
- Python 3.10, Ultralytics YOLOv8, OpenCV, Flask/FastAPI
- Hard Hat Workers-style annotated dataset (helmet / head / person)
- Wi-Fi coverage across the monitored zones

08. Expected Outcomes

- A working multi-node PPE monitoring demonstrator with live annotated dashboard.
- A fine-tuned YOLOv8s detector with mAP, precision and recall measured on the student's own splits and documented in the report.
- A violation log with evidence frames and daily compliance trend charts.

- A complete documentation set: report, PPT, viva Q&A and deployment guide.

09. Applications / Use Cases

- Demonstrating automated safety compliance for construction-site case studies
- Teaching material for object detection training, mAP evaluation and embedded camera streaming
- Reference build for other PPE classes (vests) via the documented extension path
- Prototype compliance-assist tool for site-safety demonstrations (not a certified safety system)

10. Future Scope

- Safety-vest and multi-class PPE detection on extended datasets
- On-device alerting (buzzer/SMS) wired to high-confidence violations
- PTZ or multi-angle nodes to reduce occlusion misses
- Active-learning loop: reviewer-corrected frames folded back into fine-tuning